

Carlos Vinicios Martins Rocha
Matrícula: 2212893

**Projeto Final de Programação: Ferramenta para
Visualização e Anotação de Curvas de Poços de
Petróleo.**

Rio de Janeiro, Brasil

Julho - 2023

Carlos Vinicios Martins Rocha
Matrícula: 2212893

**Projeto Final de Programação: Ferramenta para
Visualização e Anotação de Curvas de Poços de Petróleo.**

Trabalho apresentado ao coordenador do Programa de Pós-Graduação em Informática da PUC-Rio como requisito para obtenção de nota na disciplina INF2102 - Projeto Final de Programação.

Departamento de Informática
Pontifícia Universidade Católica do Rio de Janeiro

Orientador: Prof. Dr. Hélio Côrtes Vieira Lopes

Rio de Janeiro, Brasil
Julho - 2023

Sumário

1	ESPECIFICAÇÃO DO PROJETO	3
1.1	Finalidade	3
1.2	Escopo	3
1.3	Requisitos	4
1.3.1	Requisitos Funcionais	4
1.3.2	Requisitos Não-Funcionais	4
2	ARQUITETURA E PROJETO	6
2.1	Descrição das etapas da aplicação	7
3	DOCUMENTAÇÃO PARA O USUÁRIO	14
3.1	Carregando uma Base de Dados	15
3.2	Visualizando os Dados e Fazendo Anotações	15
3.3	Exportando os Dados Anotados	18
3.4	Removendo uma Base de Dados	18
4	CONSIDERAÇÕES FINAIS	20

1 Especificação do Projeto

1.1 Finalidade

Esse documento apresenta as especificações de um sistema de visualização e anotação de curvas de poços de petróleo, com o intuito de auxiliar pesquisadores na exploração de bases de dados e no desenvolvimento de modelos de aprendizado de máquina.

O recém crescimento da área de Ciência de Dados (Data Science) tem como propósito transformar informações em dados e conhecimento em informação. Projetos de Ciência de Dados buscam aprimorar processos e auxiliar na tomada de decisões embasadas em dados, resultando em um aumento de valor para eles.

Em um cenário onde os dados são amplamente gerados, eles se tornam um recurso crucial para melhoria do modelo de negócio e para otimização dos resultados. Entretanto, para o desenvolvimento de certos modelos de aprendizados de máquina, é necessário que um especialista do domínio realize a inspeção visual e o rotulamento dos dados, conhecido como treinamento supervisionado.

Portanto, como resultado do trabalho espera-se disponibilizar uma ferramenta em que os especialistas possam realizar suas inspeções visuais, assim como, criar os rótulos necessários para o treinamento do modelo projetado por ele. Além disso, espera-se que o programa também possa auxiliar outras equipes no desenvolvimento de projetos que envolvam a criação de bases de dados para treinamentos supervisionados.

1.2 Escopo

O escopo deste projeto é a produção de um sistema de visualização e anotação de curvas de poços de petróleo. O programa deve auxiliar o usuário a visualizar os dados presentes na base importada, em forma de série temporal. Além da visualização, o programa deve permitir que o usuário possa fazer anotações categóricas e binária. As anotações realizadas podem ser exportadas em formato ".csv", produzindo assim, bases anotadas para treinamento de modelos supervisionados.

1.3 Requisitos

1.3.1 Requisitos Funcionais

O requisitos funcionais do sistemas foram elicitados baseado na finalidade geral do programa, permitir visualização e à anotação de curvas de poços de petróleo.

- **RF01** - A ferramenta deve permitir que o usuário carregue um arquivo ".CSV" de qualquer diretório do seu computador;
- **RF02** - A ferramenta deve permitir que o usuário visualize as informações do arquivo selecionado durante a importação de dados;
- **RF03** - A ferramenta deve permitir que o usuário selecione dentre os arquivos carregados, um arquivo que será utilizado para visualização e anotação de dados;
- **RF04** - A ferramenta deve permitir que o usuário cadastre classes e suas subcategorias para realizar anotações;
- **RF05** - A ferramenta deve permitir que o usuário selecione dentre as classes cadastradas, a classe referente ao conjunto de pontos selecionados;
- **RF06** - A ferramenta deve permitir que o usuário crie gráficos a partir das variáveis presentes no arquivo selecionado;
- **RF07** - A ferramenta deve permitir que o usuário selecione pontos específicos para realizar uma anotação, exibindo para ele quais foram selecionados;
- **RF08** - A ferramenta deve exibir os pontos anotados para o usuário, assim que as variáveis e classes forem selecionadas;
- **RF09** - A ferramenta deve permitir que o usuário exporte as anotações realizadas para qualquer diretório do seu computador, bem como escolher quais atributos deseja exportar;
- **RF10** - A ferramenta deve permitir que o usuário remova um arquivo carregado, excluindo todos os dados presentes na ferramenta;

1.3.2 Requisitos Não-Funcionais

Dentre os requisitos não-funcionais que a ferramenta deve apresentar, listam-se:

- **RNF01 - Performance:** Refere-se ao tempo de resposta durante o uso das funcionalidades fornecidas pelo sistema;
- **RNF02 - Usabilidade:** Refere-se à facilidade de aprendizagem e uso do sistema;

- **RNF03 - Confiabilidade:** Deve-se buscar baixa taxa de erro durante o uso do sistema e robustez para resolvê-los;
- **RNF04 - Padrões:** Refere-se à conformidade do desenvolvimento do sistema com padrões de software em geral;
- **RNF05 - Compatibilidade:** Refere-se à compatibilidade de uso do sistema em OS Windows e Linux;

2 Arquitetura e Projeto

O projeto foi desenvolvido utilizando o *framework* Electron, que permite a criação de aplicações desktop, multi plataforma, utilizando a linguagem de programação JavaScript. Para a interface, foi utilizado o ReactJS, outro *framework* que utiliza JavaScript para criação de visuais baseados em componentes. Para a criação dos gráficos, foram utilizadas as bibliotecas Plotly e DanfoJS, sendo a segunda também responsável pelo gerenciamento das bases carregadas para o programa. Como ambiente de execução para todos os *framework* listados, escolheu-se o NodeJS na versão 16.15.1.

```
├── databases
│   └── WELL-XYZ.csv
├── databasesConfig
│   └── WELL-XYZ.json
├── public
├── src
│   ├── App.css
│   ├── App.js
│   ├── components
│   │   ├── AlertDialog
│   │   ├── AnnotationClass
│   │   ├── AnnotationExport
│   │   ├── DatabaseItem
│   │   ├── MultiValueTextField
│   │   ├── UploadDatabase
│   │   └── VariableItem
│   ├── electron-starter.js
│   ├── electron-wait-react.js
│   ├── index.css
│   ├── index.js
│   ├── pages
│   │   ├── DatabaseAnnotation
│   │   └── mainWindow
│   ├── Preload.js
│   ├── routes
│   ├── __tests__
│   └── utils
```

A árvore de arquivos acima descreve a estrutura de diretórios do sistema, apresentando em sua raiz os subdiretórios: *databases*, que será utilizado em tempo de execução para armazenar as bases de dados carregadas pelo usuário; *databasesConfig*, encarregado de armazenar as configurações particulares de cada uma das bases; *public*, que persiste os arquivos necessários para renderização da interface gráfica do ReactJS; *src*, que contém toda a lógica para criação dos componentes da interface e suas interações.

Além dos componentes, o diretório *src* também contém os arquivos que controlam as funcionalidades do Electron, como a sua comunicação com as chamadas ao sistema operacional, interface de comunicação com ReactJS e gerenciamento do ciclo de vida da aplicação. Dentro do *src*, existe a pasta `__tests__`, nela está especificado alguns testes de unidade para verificação da integridade dos componentes desenvolvidos para a aplicação.

2.1 Descrição das etapas da aplicação

O controle inicial da aplicação é realizado pela *mainWindow*, que se encontra dentro do diretório *src/pages*. Nesta janela, podemos importar uma nova base para o sistema (Figura 1), remover (Figura 2) ou editar bases anteriormente carregadas. Quando o usuário selecionar a opção de carregar base, o componente *UploadDatabase* será chamado, apresentando as colunas presentes no arquivo e disponibilizando uma região para construção de um gráfico com as colunas.

A *DatabaseAnnotation* fica responsável por controlar o processo de visualização através do gráfico gerado de acordo com as variáveis selecionadas pelo usuário, como mostra o fluxo da Figura 3. Nesta tela, o usuário também realizará suas anotações através da adição de classes e seleção dos pontos alvos, por meio de interação com o gráfico renderizado.

Com relação as classes, o programa dispõe de dois tipos específicos: booleano e categórico. Quando o usuário opta por adicionar uma classe do tipo "categórico", o programa irá solicitar o preenchimento das categóricas que compõem a classe informada. Para o booleano, o sistema irá preencher automaticamente as categorias com os valores de 0 e 1. Todo esse processo está descrito na Figura 4.

Para a anotação de um ou mais pontos, o usuário irá interagir com as ferramentas disponíveis no canto superior direito do gráfico. Após a escolha da ferramenta de seleção, o usuário selecionará os pontos desejados e os mesmos serão exibidos ao lado do gráfico. Por fim, ele escolherá a categoria desejada e salvará as suas anotações, como demonstrado na Figura 5.

Tendo em vista que o sistema apenas serve para visualização e anotação das bases, ele conta com a funcionalidade de exportar as anotações realizadas pelo usuário. Tal

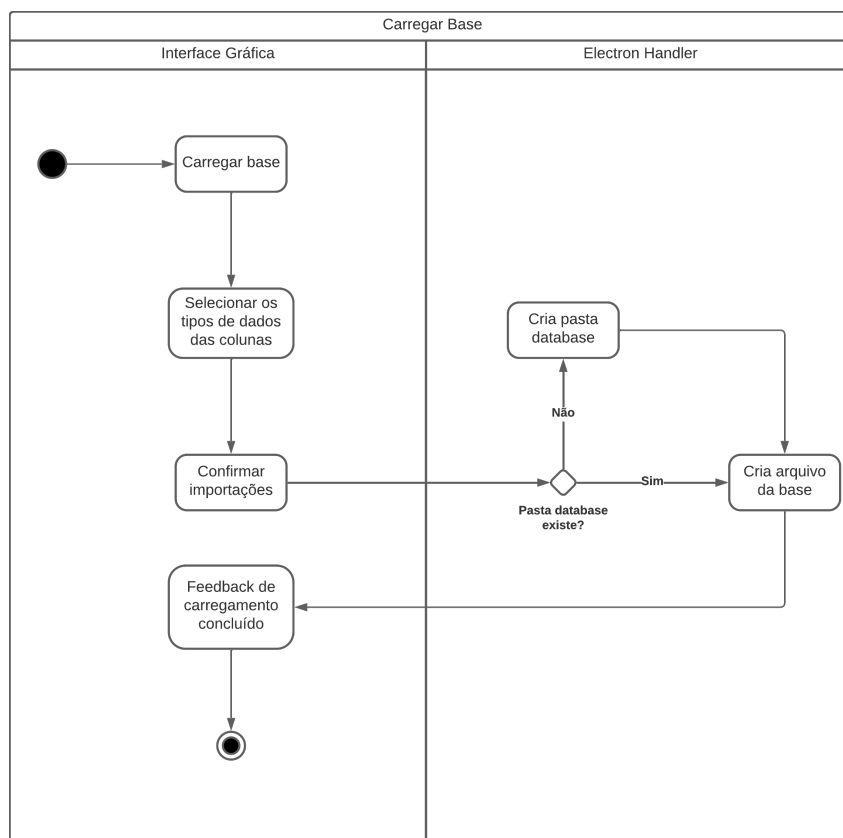


Figura 1 – Diagrama de Atividade do Fluxo de Adição de Base.

funcionalidade também é acessada pela tela *DatabaseAnnotation*, que irá expor ao usuário as colunas aptas a exportação e solicitará a escolha do caminho para exportação da base anotada, como exibido no fluxo de atividade da Figura 6.

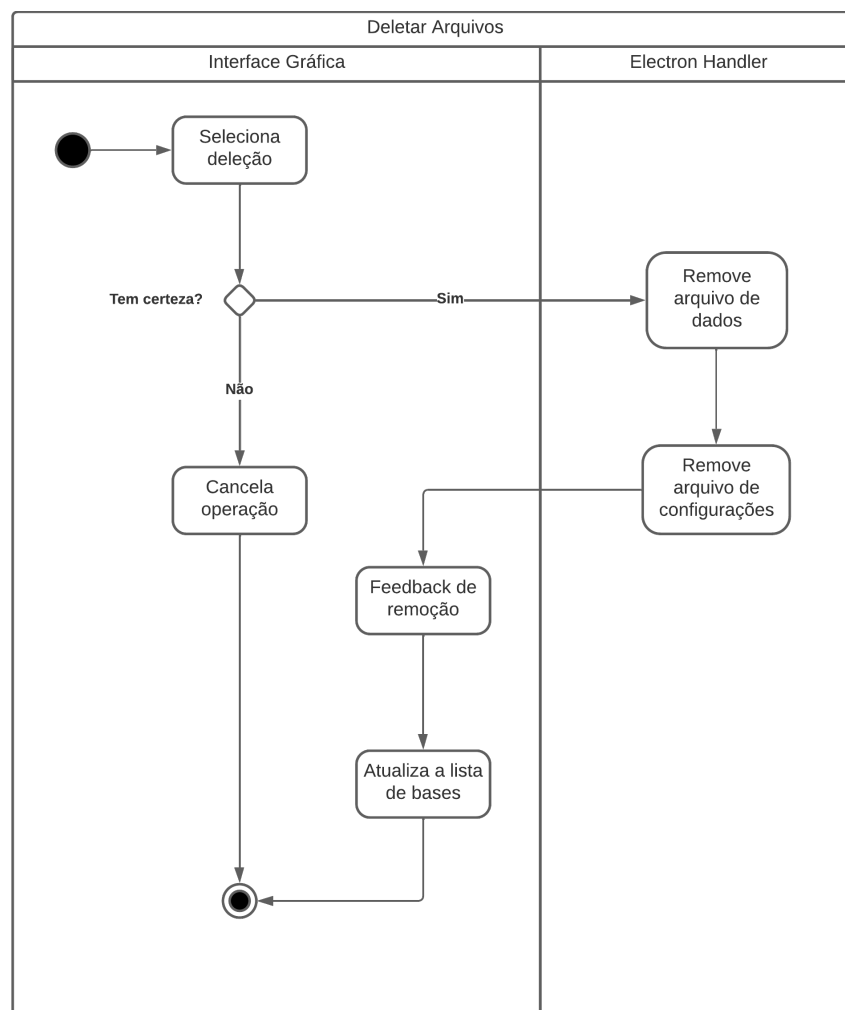


Figura 2 – Diagrama de Atividade do Fluxo de Remoção da Base.

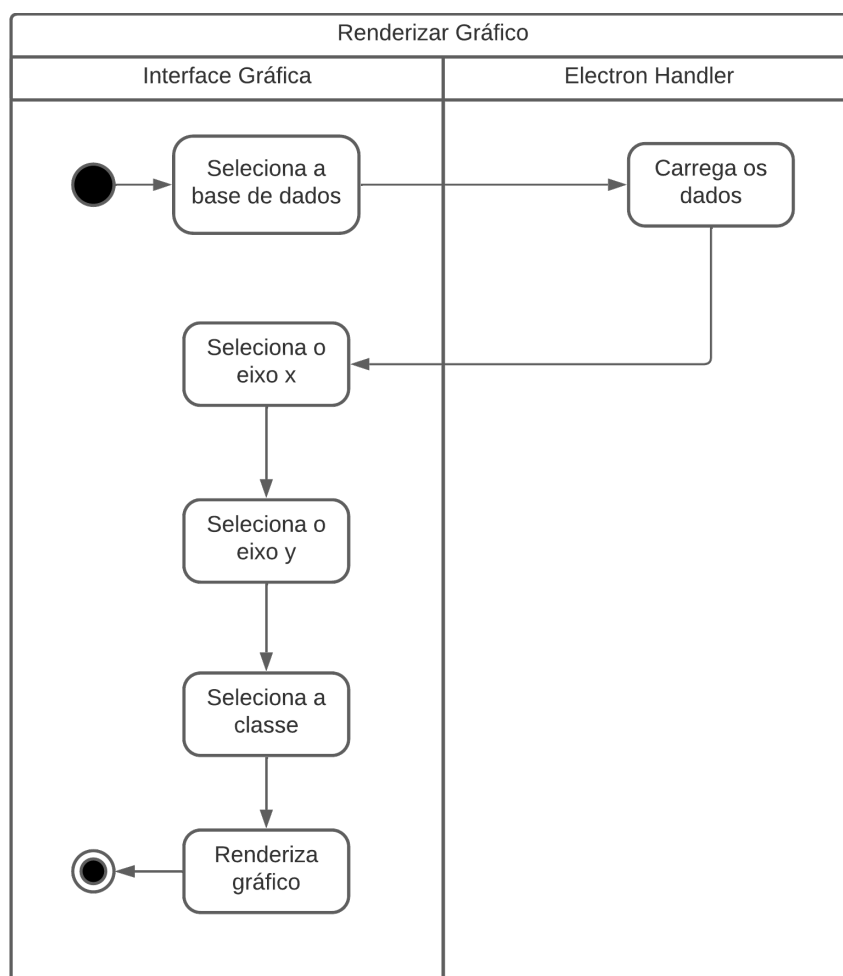


Figura 3 – Diagrama de Atividade do Fluxo de Renderização do Gráfico.

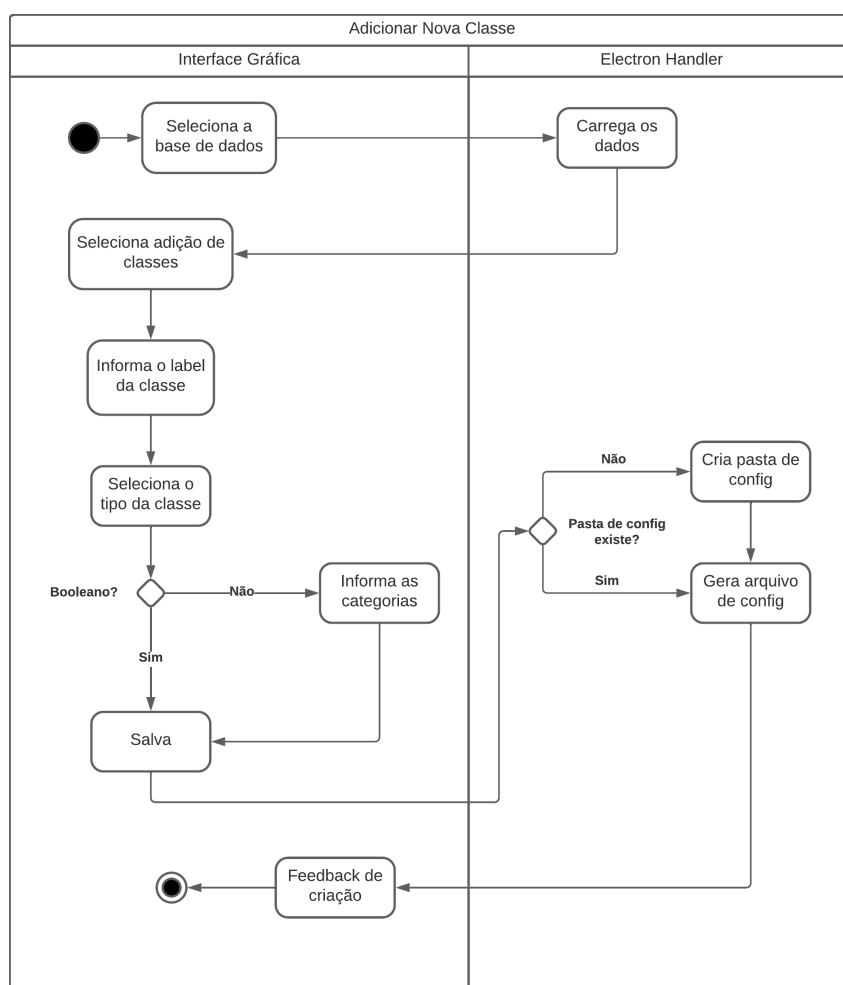


Figura 4 – Diagrama de Atividade do Fluxo de Adição de uma Nova Classe.

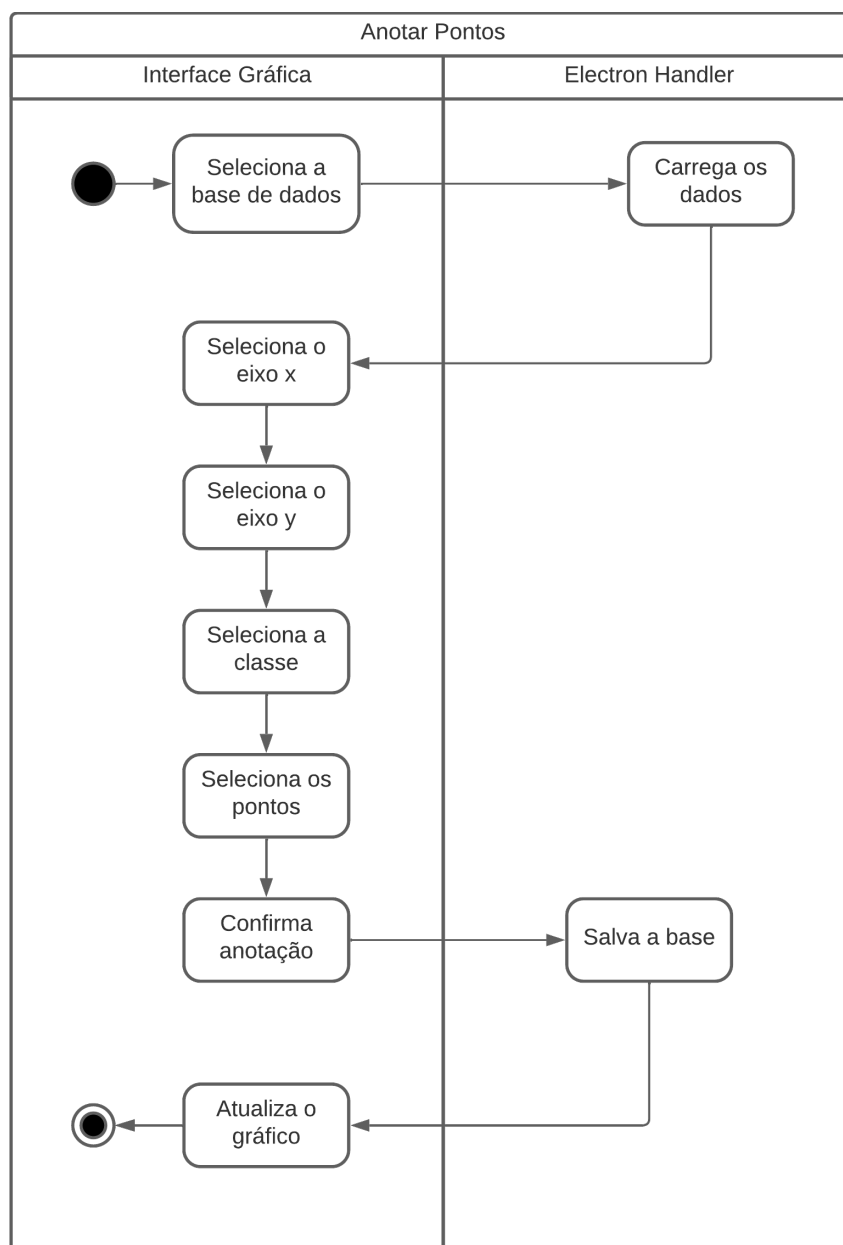


Figura 5 – Diagrama de Atividade do Fluxo de Anotação de Pontos.

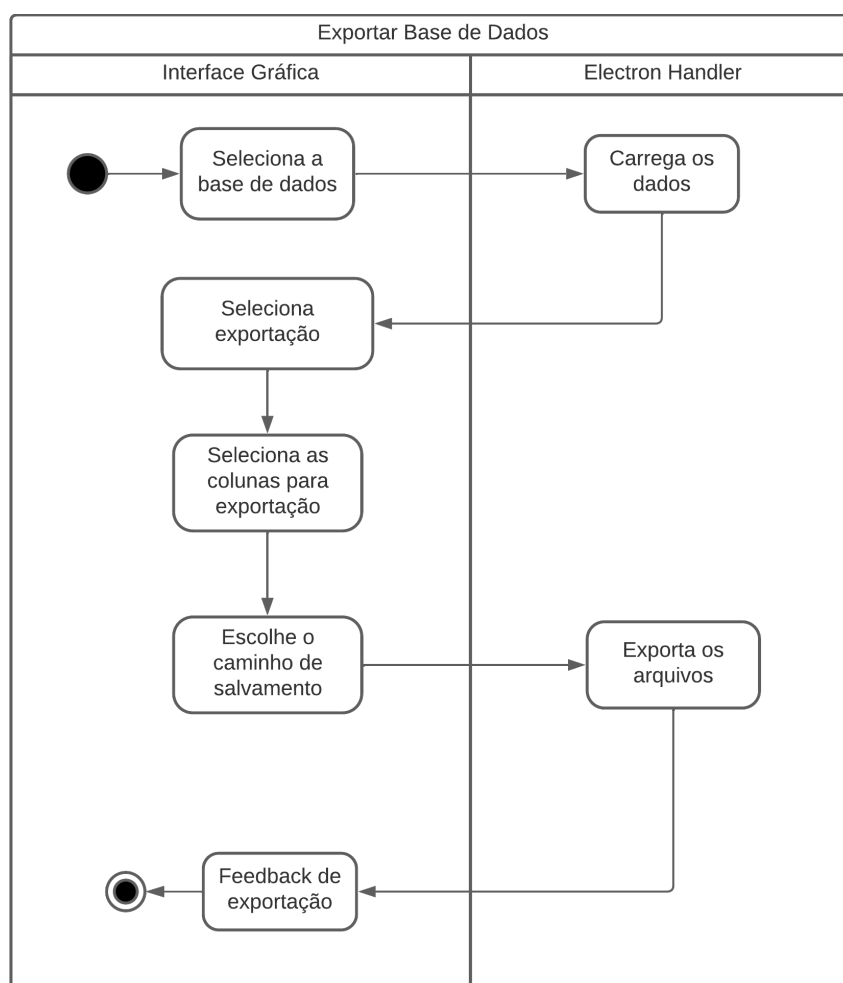


Figura 6 – Diagrama de Atividade do Fluxo de Exportação da Base de Dados.

3 Documentação para o Usuário

A aplicação desenvolvida tem como principal objetivo, permitir que seus usuários possam visualizar as séries temporais de curvas de poços de petróleo, além de permitir que possam realizar anotações de forma pontual nos dados que foram carregados. Os principais usuários pensados foram os pesquisadores do projeto de pesquisa e desenvolvimento SIARP, que buscam desenvolver métodos automatizados para monitoramento de danos em poços de petróleo. Dentre os métodos automatizados, estão presentes os algoritmos de aprendizado de máquina supervisionados.

Para o gerenciamento das dependências e códigos do projeto, espera-se que o usuário tenha um conhecimento prévio em NodeJS e Git. O primeiro se faz necessário para execução do projeto e pode ser instalado através do NVM ¹, que facilita a instalação de versões do NodeJS ², criando ambientes virtuais de forma isolada, evitando conflito entre projetos.

No que diz respeito ao Git, o usuário deverá realizar um clone do projeto. Para isso, se faz necessário ter o git instalado em sua máquina. O mesmo pode ser instalado através de diversas distribuições ³.

Após o clone do projeto, o usuário deve instalar os pacotes necessários, presentes no arquivo package.json, através do comando: **npm install**. Ao finalizar o processo de instalação, o usuário deve executar o comando: **npm run dev**, para que a aplicação seja iniciada e a tela principal do programa seja exibida, Figura 7.

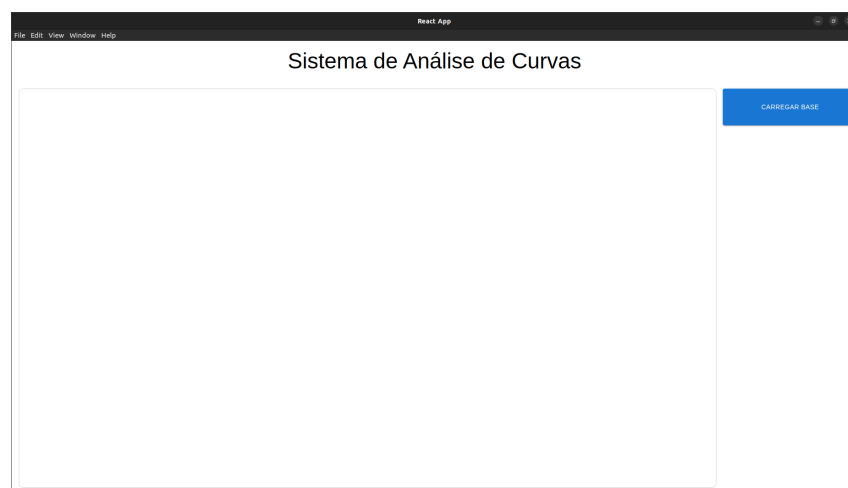


Figura 7 – Tela Principal da Aplicação.

¹ <https://github.com/nvm-sh/nvm>

² Para este projeto recomenda-se que o usuário utilize a versão 16.15.1.

³ Exemplo de instalação do Git: <https://git-scm.com/downloads>

3.1 Carregando uma Base de Dados

Para que o usuário carregue uma nova base no sistema, ele deve clicar no botão "Carregar Base", que exibirá um seletor de arquivo, permitindo que navegue até arquivo ".csv" desejado, como ilustrado na Figura 7. Após a escolha e confirmação do arquivo, uma caixa de diálogo será exibida (Figura 8), apresentando os atributos, e seus tipos de dados, presentes na base escolhida. Nessa caixa de diálogo, o usuário também tem a opção de criar um gráfico simples, para visualizar a exatidão dos atributos e fazer alterações nos tipos de dados.

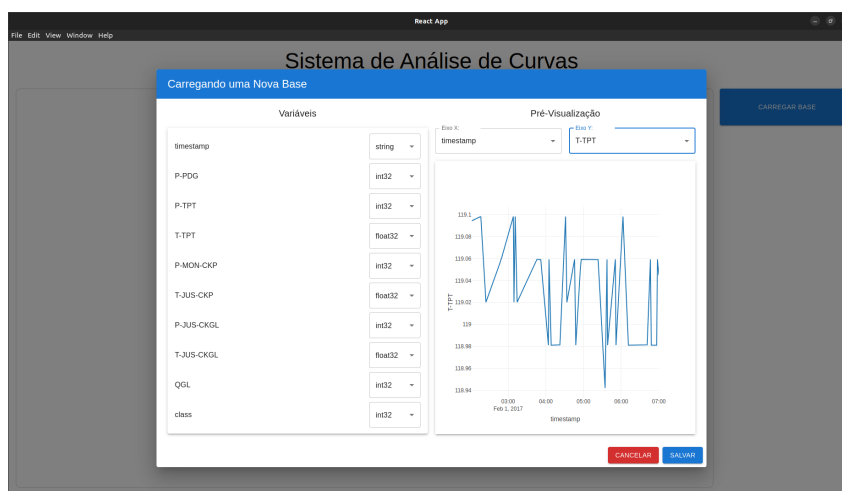


Figura 8 – Caixa de Diálogo da Importação de Arquivo.

3.2 Visualizando os Dados e Fazendo Anotações

Após o usuário ter realizado o processo de carregamento dos dados para o sistema, a base ficará listada na página principal, podendo ser acessada através do botão esquerdo do mouse, como ilustra a Figura 9. Selecionado uma base, uma nova tela será mostrada, solicitando que o usuário escolha qual variável deseja utilizar para o eixo x e y do gráfico e qual classe deseja dar foco no gráfico a ser renderizado, como demonstra a Figura 10.

Caso seja a primeira vez que o usuário utilize o sistema, é necessário que o mesmo cadastre uma nova classe de anotação. Para isso, basta que ele clique no ícone "lápiz", situado ao lado do seletor de classe, para que o formulário de cadastro de classe seja exibido na tela (Figura 11). O formulário de classe irá pedir que o usuário informe um nome para classe e selecione o seu tipo. Caso o tipo seja categórico, um novo campo textual será exibido, permitindo que o usuário entre com as subcategorias existentes para as classes, informando-as de forma contínua separadas por espaço em branco.

O formulário de classes também permite que o usuário remova ou edite uma classe

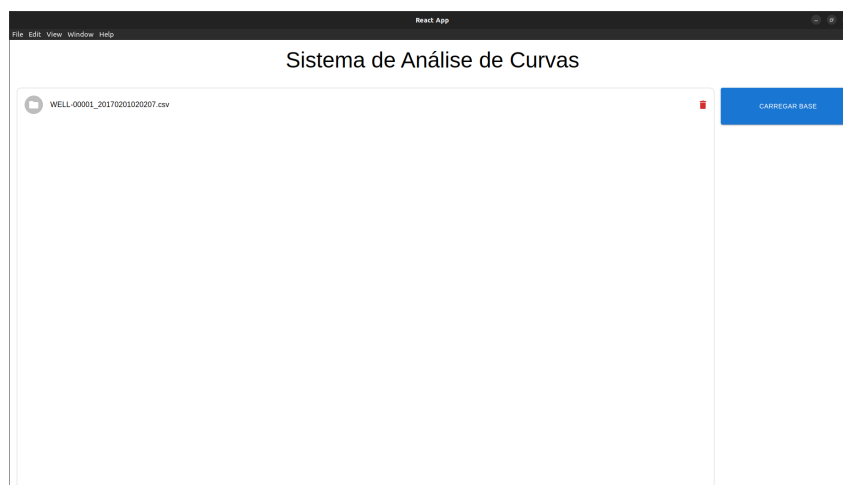


Figura 9 – Tela Principal da Aplicação com Base Carregada.

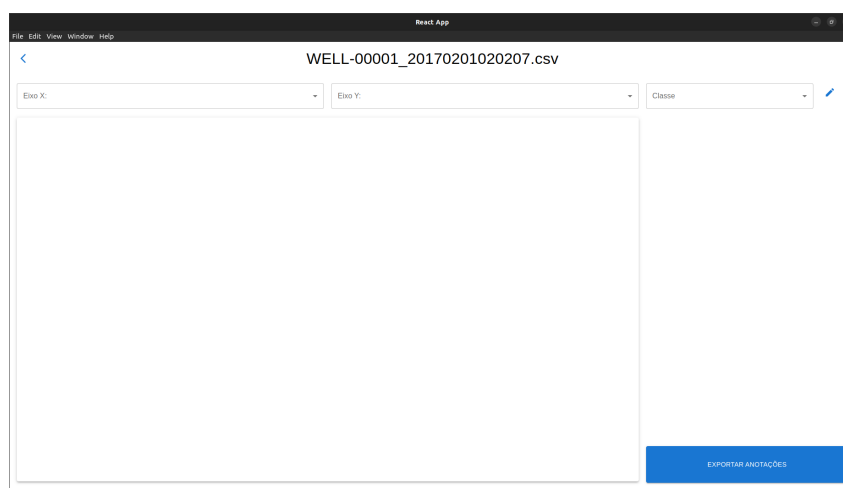


Figura 10 – Tela de Anotação de Dados.

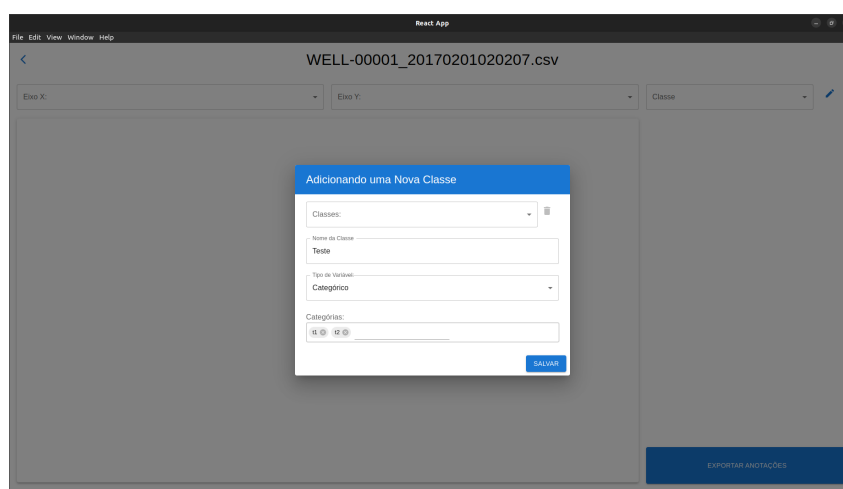


Figura 11 – Formulário de Adição de Classes.

que não esteja em uso⁴. Para remoção, basta o usuário selecionar a classe desejada na primeira lista de seleção e clicar no ícone da lixeira ao lado dela, como demonstrado na Figura 12. A edição ocorre de forma similar, através da seleção da classe desejada, em conjunto com as modificações desejadas nos campos exibidos em tela.

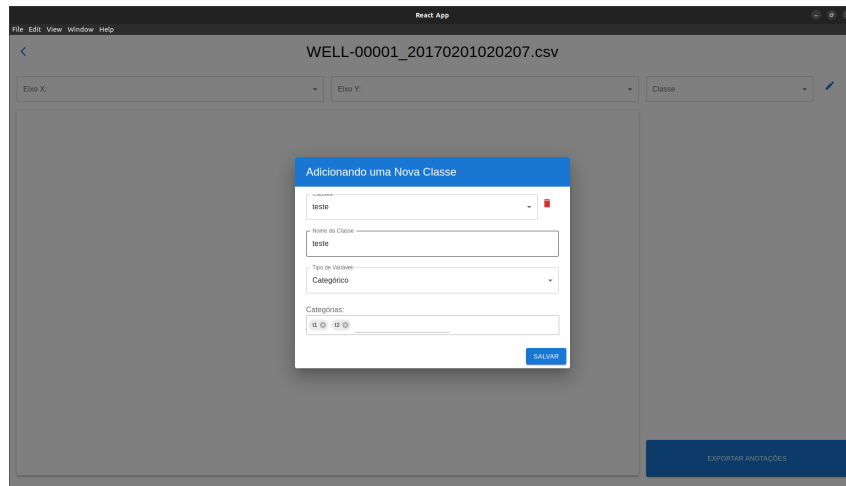


Figura 12 – Remoção de Classe.

Realizadas as etapas para renderização do gráfico, descritas anteriormente, o usuário pode dar início à anotação dos dados. A seleção dos dados é feita através da interação com o motor de renderização, que fornece certas opções aos usuários como: zoom, seleção em caixa, seleção livre, baixar o gráfico, etc. (Figura 13).



Figura 13 – Ferramentas do Plotly.

Escolhida a opção de seleção de sua preferência, o usuário irá selecionar os pontos que deseja anotar. Após a seleção, os pontos serão exibidos em uma lista ao canto direito

⁴ Considera-se a classe em uso, quando o gráfico principal esteja utilizando ela em sua etapa de renderização

do gráfico, permitindo que o usuário realize uma validação visual. Caso esteja tudo certo, basta clicar no botão de salvar, para que as anotações sejam persistidas no sistema, como exibido na Figura 14.

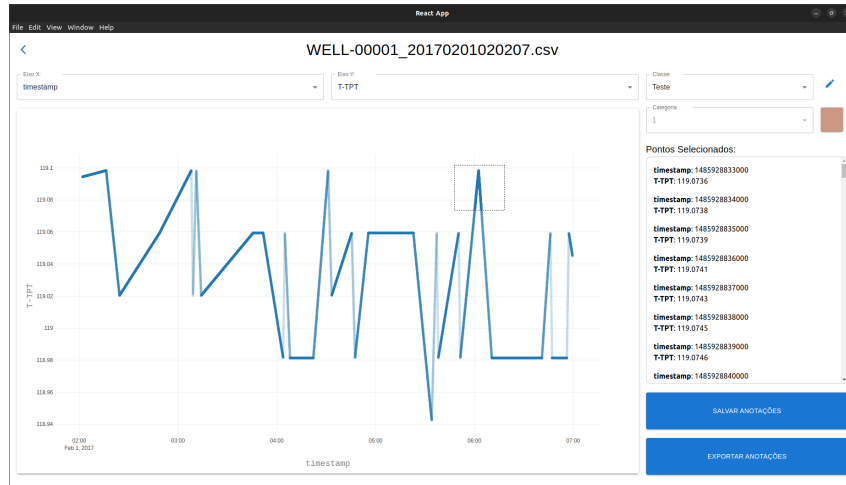


Figura 14 – Lista de Pontos Seleccionados.

3.3 Exportando os Dados Anotados

A exportação da base de dados é realizada na mesma tela de anotações dos pontos, desta forma, caso o usuário esteja na página principal, é necessário que selecione o arquivo que deseja exportar. Na tela de anotações o usuário vai encontrar no canto inferior direito, o botão que realiza a exportação dos dados. Ao clicar no botão, uma breve parte da base de dados será exibida na tela, permitindo que o usuário escolha quais colunas deseja exportar através das *checkbox* presentes ao lado de cada uma delas (Figura 15). Finalizado a seleção das colunas, o usuário irá clicar no botão de exportar, exibindo o seletor de arquivos, para que escolha o caminho de criação do arquivo exportado.

3.4 Removendo uma Base de Dados

Voltando a página principal, o usuário tem a opção de remover um arquivo carregado, ao clicar no ícone "lixeira" que se encontra ao lado direito de todas as bases carregadas (Figura 9). Antes de realizar a exclusão completa dos dados, o sistema irá exibir uma caixa de confirmação (Figura 16), com o intuito de evitar remoções acidentais.

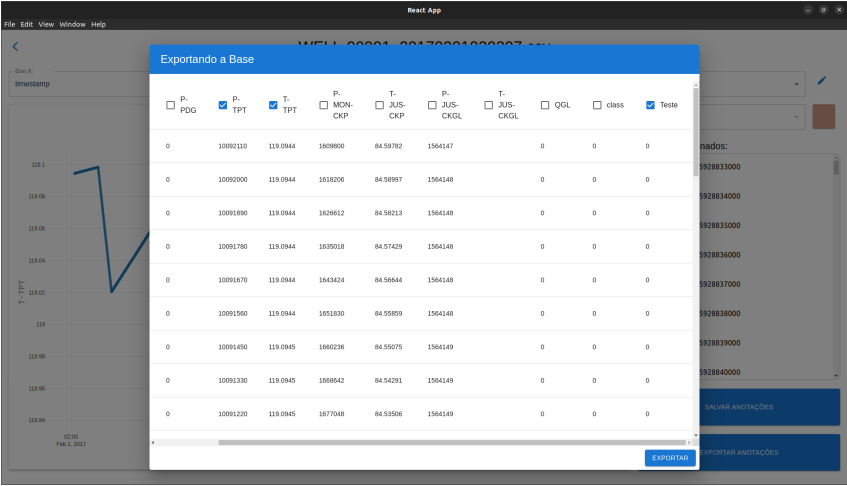


Figura 15 – Lista de Pontos Selecionados.

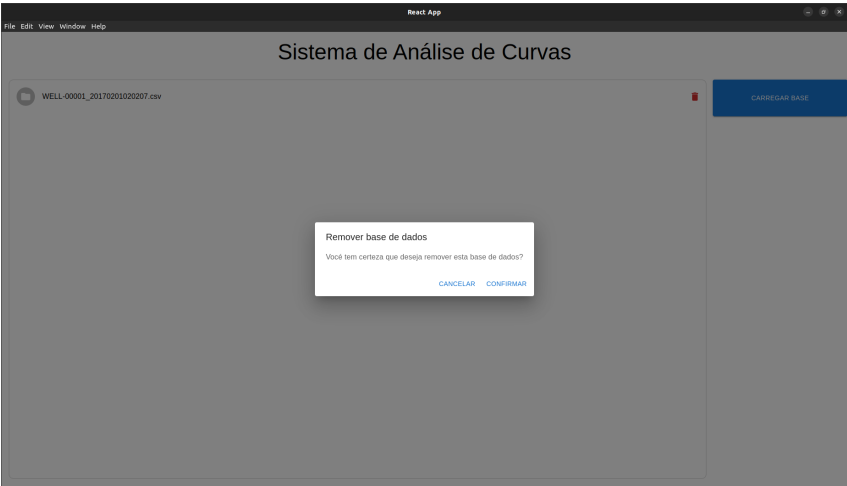


Figura 16 – Caixa de Confirmação da Remoção.

4 Considerações Finais

Ao fim do desenvolvimento deste projeto, conclui-se que, além do desenvolvimento e aperfeiçoamento de habilidades técnicas pessoais, a ferramenta idealizada irá auxiliar na evolução do projeto de pesquisa e desenvolvimento do SIARP. A ferramenta traz como sua principal contribuição a possibilidade de anotar conjuntos de dados de maneira rápida, reduzindo o tempo gasto para criação das bases utilizadas para treinamento dos nossos modelos de aprendizado de máquina. Além disso, a ferramenta também permite que os usuários possam criar gráficos interativos de maneira simples e rápida, ajudando no processo de inspeção visual nos dados.

Após realizado todos os passos contidos no manual do usuário, o sistema foi capaz de auxiliar os pesquisadores do SIARP no desenvolvimento de uma base anotada de anomalias no conjunto de dados. Essas anomalias tratam-se de pontos afastados da curva padrão, portanto, são anomalias visualmente perceptíveis. Ao fim do processo de anotação, os dados foram exportados e carregados em *Jupyter Notebook* para que as etapas seguintes da pesquisa fossem realizadas.

Seguindo os mesmos passos, a ferramenta permite que os pesquisadores possam anotar anomalias em janelas de tempo, relacionadas a algum acontecimento específico da base estudada. Essa tarefa pode ser realizada através do uso das classes categóricas, possibilitando uma melhor especificação da região de interesse.

Com relação aos percalços, durante os testes foram encontrados problemas de compatibilidade com a versão do Linux CentOS e de disposição das informações em telas menores. O primeiro problema está relacionado a ausência do Electron (dependência obrigatória) para a distribuição do CentOS, impossibilitando o funcionamento da ferramenta. Para contornar esse problema, é necessário um maior estudo do CentOS em busca do ponto de incompatibilidade, o que gerará uma forma diferente de compilar e executar a aplicação.

Por último, o problema da disposição das informações pode ser resolvido através da melhoria da responsividade da tela. Para isso, é necessário que o sistema seja testado em diferentes monitores com variadas resoluções, com o objetivo de mapear os ajustes necessários evitando a quebra do layout.